INEXACT TENSOR METHODS WITH DYNAMIC ACCURACIES

Nikita Doikov, Yurii Nesterov

REPRINT | 3309

UCLouvain

CORE | LIDAM
Louvain Institute of Data Analysis and Modeling in economics and statistics

Inexact Tensor Methods with Dynamic Accuracies

Nikita Doikov [1] **Yurii Nesterov** [2]

Abstract

In this paper, we study inexact high-order Tensor Methods for solving convex optimization problems with composite objective. At every step of such methods, we use approximate solution of the auxiliary problem, defined by the bound for the residual in function value. We propose two dynamic strategies for choosing the inner accuracy: the first one is decreasing as $1/k^{p+1}$, where $p \geq 1$ is the order of the method and k is the iteration counter, and the second approach is using for the inner accuracy the last progress in the target objective. We show that inexact Tensor Methods with these strategies achieve the same global convergence rate as in the error-free case. For the second approach we also establish local superlinear rates (for $p \geq 2$), and propose the accelerated scheme. Lastly, we present computational results on a variety of machine learning problems for several methods and different accuracy policies.

1. Introduction

1.1. Motivation

With the growth of computing power, high-order optimization methods are becoming more and more popular in machine learning, due to their ability to tackle the ill-conditioning and to improve the rate of convergence. Based on the work (Nesterov & Polyak, 2006), where global complexity guarantees for the Cubic regularization of Newton method were established, a significant leap in the development of second-order optimization algorithms was made, discovering stochastic and randomized methods (Kohler & Lucchi, 2017; Tripuraneni et al., 2018; Cartis & Scheinberg, 2018; Doikov & Richtárik, 2018; Wang et al., 2018; Zhou et al., 2019), which have better convergence rate, than the corresponding first-order analogues. The main weakness, though, is that every step of Newton method is much more expensive. It requires to solve the subproblem, which is a minimization of quadratic function with a regularizer, and possibly with some additional nondifferentiable components. Therefore, the idea to employ *higher* derivatives into optimization schemes was remaining questionable, because of the high cost of the computations. However, recently (Nesterov, 2019a) it was shown, that the third-order Tensor Method for convex minimization problems admits very effective implementation, with the cost, which is comparable to that of the Newton step.

Now we have a family of methods (starting from the methods of order one), for each iteration of which may need to call some auxiliary subsolver. Thus, it becomes important to study: *which level of exactness we need to ensure at the step for not loosing the fast convergence of the initial method*. In this work, we suggest to describe approximate solution of the subproblem in terms of the residual in function value. We propose two strategies for the inner accuracies, which are *dynamic* (changing with iterations). Indeed, there is no need to have a very precise solution of the subproblem at the first iterations, but we reasonably ask for higher precision closer to the end of the optimization process.

1.2. Related Work

Global convergence of the first-order methods with inexact proximal-gradient steps was studied in (Schmidt et al., 2011). The authors considered the errors in the residual in function value of the subproblem, and require them to decrease with iterations at an appropriate rate. This setting is the most similar to the current work.

In (Cartis et al., 2011a;b), adaptive second-order methods with cubic regularization and inexact steps were proposed. High-order inexact tensor methods were considered in (Birgin et al., 2017; Jiang et al., 2018; Grapiglia & Nesterov, 2019c;d; Cartis et al., 2019; Lucchi & Kohler, 2019). In all of these works, the authors describe approximate solution of the subproblem in terms of the corresponding first-order optimality condition (using the gradients). This can be difficult to achieve by the current optimization schemes, since

[1]Institute of Information and Communication Technologies, Electronics and Applied Mathematics (ICTEAM), Catholic University of Louvain (UCL), Belgium [2]Center for Operations Research and Econometrics (CORE), Catholic University of Louvain (UCL), Belgium. Correspondence to: Nikita Doikov <nikita.doikov@uclouvain.be>, Yurii Nesterov <yurii.nesterov@uclouvain.be>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

it is more often that we have a better (or the only) guarantees for the decrease of the residual in function value. The latter one is used as a measure of inaccuracy in the recent work (Nesterov, 2019b) on the inexact Basic Tensor Methods. However, only the constant choice of the accuracy level is considered there.

1.3. Contributions

We propose new dynamic strategies for choosing the inner accuracy for the general Tensor Methods, and several inexact algorithms based on it, with proven complexity guarantees, summarized next (we denote by δ_k the required precision for the residual in function value of the auxiliary problem):

- The rule $\delta_k := 1/k^{p+1}$, where $p \geq 1$ is the order of the method, and k is the iteration counter.

  Using this strategy, we propose two optimization schemes: Monotone Inexact Tensor Method I (Algorithm 1) and Inexact Tensor Method with Averaging (Algorithm 3). Both of them have the global complexity estimates $O(1/\varepsilon^{\frac{1}{p}})$ iterations for minimizing the convex function up to ε -accuracy (see Theorem 1 and Theorem 5). The latter method seems to be the first *primal* high-order scheme (aggregating the points from the primal space only), having the explicit distance between the starting point and the solution, in the complexity bound.

- The rule $\delta_k := c \cdot (F(x_{k-2}) - F(x_{k-1}))$, where $F(x_i)$ are the values of the target objective during the iterations, and $c \geq 0$ is a constant.

  We incorporate this strategy into our Monotone Inexact Tensor Method II (Algorithm 2). For this scheme, for minimizing convex functions up to ε-accuracy by the methods of order $p \geq 1$, we prove the global complexity proportional to $O(1/\varepsilon^{\frac{1}{p}})$ (Theorem 2). The global rate becomes linear, if the objective is uniformly convex (Theorem 3).

  Assuming that $\delta_k := c \cdot (F(x_{k-2}) - F(x_{k-1}))^{\frac{p+1}{2}}$, for the methods of order $p \geq 2$ as applied to minimization of strongly convex objective, we also establish the local superlinear rate of convergence (see Theorem 4).

- Using the technique of Contracting Proximal iterations (Doikov & Nesterov, 2019a), we propose inexact Accelerated Scheme (Algorithm 4), where at each iteration k, we solve the corresponding subproblem with the precision $\zeta_k := 1/k^{p+2}$ in the residual of the function value, by inexact Tensor Methods of order $p \geq 1$. The resulting complexity bound is $\tilde{O}(1/\varepsilon^{\frac{1}{p+1}})$ inexact tensor steps for minimizing the convex function up to ε accuracy (Theorem 6).

- Numerical results with empirical study of the methods for different accuracy policies are provided.

1.4. Contents

The rest of the paper is organized as follows. Section 2 contains notation which we use throughout the paper, and declares our problem of interest in the composite form. In Section 3 we introduce high-order model of the objective and describe a general optimization scheme using this model. Then, we summarize some known techniques for computing a step for the methods of different order. In Section 3.2 we study monotone inexact methods, for which we guarantee the decrease of the objective function for every iteration, and in Section 3.3 we study the methods with averaging. In Section 4 we present our accelerated scheme. Section 5 contains numerical results. Missing proofs are provided in the supplementary material.

2. Notation

In what follows, we denote by $\mathbb{E}$ a finite-dimensional real vector space and by $\mathbb{E}^*$ its dual space, which is a space of linear functions on $\mathbb{E}$. The value of function $s \in \mathbb{E}^*$ on $x \in \mathbb{E}$ is denoted by $\langle s, x \rangle$. One can always identify $\mathbb{E}$ and $\mathbb{E}^*$ with $\mathbb{R}^n$, when some basis is fixed, but often it is useful to seperate these spaces, in order to avoid ambiguities.

Let us fix some symmetric positive definite linear operator $B : \mathbb{E} \to \mathbb{E}^*$ and use it to define Euclidean norm for the primal variables: $\|x\| \stackrel{\text{def}}{=} \langle Bx, x \rangle^{1/2}$, $x \in \mathbb{E}$. Then, the norm for the dual space is defined as:

$$\|s\|_* \stackrel{\text{def}}{=} \max_{h \in \mathbb{E}: \, \|h\| \leq 1} \langle s, x \rangle \; = \; \langle s, B^{-1} s \rangle^{1/2}, \; s \in \mathbb{E}^*.$$

For a smooth function f, its gradient at point x is denoted by $\nabla f(x)$, and its Hessian is $\nabla^2 f(x)$. Note that for $x \in \operatorname{dom} f \subseteq \mathbb{E}$ we have $\nabla f(x) \in \mathbb{E}^*$, and $\nabla^2 f(x) h \in \mathbb{E}^*$ for $h \in \mathbb{E}$.

For $p \geq 1$, we denote by $D^p f(x)[h_1, \cdots, h_p]$ pth directional derivative of f along directions $h_1, \ldots, h_p \in \mathbb{E}$. If $h_i = h$ for all $1 \leq i \leq p$, the shorter notation $D^p f(x)[h]^p$ is used. The norm of $D^p f(x)$, which is p-linear symmetric form on $\mathbb{E}$, is induced in the standard way:

$$\begin{aligned} \|D^p f(x)\| \; &\stackrel{\text{def}}{=} \max_{\substack{h_1, \ldots, h_p \in \mathbb{E}: \\ \|h_i\| \leq 1, \; 1 \leq i \leq p}} D^p f(x)[h_1, \ldots, h_p] \\ &= \max_{h \in \mathbb{E}: \, \|h\| \leq 1} \left| D^p f(x)[h]^p \right|. \end{aligned}$$

See Appendix 1 in (Nesterov & Nemirovskii, 1994) for the proof of the last equation.

We are interested to solve convex optimization problem in

the composite form:

$$\min_{x\in\mathbb{E}}\Big\{F(x) \;\equiv\; f(x)+\psi(x)\Big\}, \qquad (1)$$

where $\psi : \mathbb{E} \to \mathbb{R}\cup\{+\infty\}$ is a *simple* proper closed convex function, and $f : \operatorname{dom}\psi \to \mathbb{R}$ is several times differentiable and convex. Basic examples of ψ which we keep in mind are: $\{0,+\infty\}$-indicator of a simple closed convex set and ℓ_1-regularization. We always assume, that solution $x^* \in \operatorname{dom}\psi$ of problem (1) does exist, denoting $F^* = F(x^*)$.

3. Inexact Tensor Methods

3.1. High-Order Model of the Objective

We assume, that for some $p \geq 1$, the pth derivative of the smooth component of our objective (1) is Lipschitz continuous.

Assumption 1 *For all $x, y \in \operatorname{dom}\psi$*

$$\|D^p f(x) - D^p f(y)\| \;\leq\; L_p\|x-y\|. \qquad (2)$$

Examples of convex functions with known Lipschitz constants are as follows.

Example 1 *For the power of Euclidean norm $f(x) = \frac{1}{p+1}\|x-x_0\|^{p+1}$, $p \geq 1$, (2) holds with $L_p = p!$ (see Theorem 7.1 in (Rodomanov & Nesterov, 2019)).*

Example 2 *For a given $a_i \in \mathbb{E}^*, 1 \leq i \leq m$, consider the log-sum-exp function:*

$$f(x) \;=\; \log\left(\sum_{i=1}^m e^{\langle a_i, x\rangle}\right), \quad x \in \mathbb{E}.$$

Then, for $B := \sum_{i=1}^m a_i a_i^*$ (assuming $B \succ 0$, otherwise we can reduce dimensionality of the problem), (2) holds with $L_1 = 1$, $L_2 = 2$ and $L_3 = 4$ (see Lemma 4 in the supplementary material).

Example 3 *Using $\mathbb{E} = \mathbb{R}$, and $a_1 = 0, a_2 = 1$ in the previous example, we obtain the logistic regression loss: $f(x) = \log(1+e^x)$.*

Let us consider Taylor's model of f around a fixed point x:

$$f(y) \;\approx\; f_{p,x}(y) \;\stackrel{\text{def}}{=}\; f(x) + \sum_{k=1}^p \tfrac{1}{k!} D^k f(x)[y-x]^k.$$

Then, from (2) we have a global bound for this approximation. It holds, for all $x, y \in \operatorname{dom}\psi$

$$|f(y) - f_{p,x}(y)| \;\leq\; \tfrac{L_p}{(p+1)!}\|y-x\|^{p+1}. \qquad (3)$$

Denote by $\Omega_H(x;y)$ the following regularized model of our objective:

$$\Omega_H(x;y) \;\stackrel{\text{def}}{=}\; f_{p,x}(y) + \tfrac{H\|y-x\|^{p+1}}{(p+1)!} + \psi(y), \qquad (4)$$

which serves as the global upper bound: $F(y) \leq \Omega_H(x;y)$, for H big enough (at least, for $H \geq L_p$). This property suggests us to use the minimizer of (4) in y, as the next point of a hypothetical optimization scheme, while x being equal to a current iterate:

$$x_{k+1} \;\in\; \operatorname{Argmin}_y \Omega_H(x_k;y), \quad k \geq 0. \qquad (5)$$

The approach of using high-order Taylor model $f_{p,x}(y)$ with its regularization was investigated first in (Baes, 2009).

Note, that for $p = 1$, iterations (5) gives the Gradient Method (see (Nesterov, 2013) as a modern reference), and for $p = 2$ it corresponds to the Newton method with Cubic regularization (Nesterov & Polyak, 2006) (see also (Doikov & Richtárik, 2018) and (Grapiglia & Nesterov, 2019a) for extensions to the composite setting).

Recently it was shown in (Nesterov, 2019a), that for $H \geq pL_p$, function $\Omega_H(x;\cdot)$ is *always convex* (despite the Taylor's polynomial $f_{p,x}(y)$ is nonconvex for $p \geq 3$). Thus, computation (5) of the next point can be done by powerful tools of Convex Optimization. Let us summarize some known techniques for computing a step of the general method (5), for different p:

- $p = 1$. When there is no composite part: $\psi(x) = 0$, iteration (5) can be represented as the Gradient Step with *preconditioning*: $x_{k+1} = x_k - \frac{1}{H}B^{-1}\nabla f(x_k)$. One can precompute inverse of B in advance, or use some numerical subroutine at every step, solving the linear system (for example, by Conjugate Gradient method, see (Nocedal & Wright, 2006)). If $\psi(x) \neq 0$, computing the corresponding prox-operator is required (see (Beck, 2017)).

- $p = 2$. One approach consists in diagonalizing the quadratic part by using eigenvalue- or tridiagonal-decomposiition of the Hessian matrix. Then, computation of the model minimizer (5) can be done very efficiently by solving some one-dimensional nonlinear equation (Nesterov & Polyak, 2006; Gould et al., 2010). The usage of fast approximate eigenvalue computation was considered in (Agarwal et al., 2017). Another way to compute the (inexact) second-order step (5) is to launch the Gradient Method (Carmon & Duchi, 2019). Recently, the subsolver based on the Fast Gradient Method with restarts was proposed in (Nesterov, 2019b), which convergence rate is $O(\frac{1}{t^6})$, where t is the iteration counter.

- $p = 3$. In (Nesterov, 2019a), an efficient method for computing the inexact third-order step (5) was outlined, which is based on the notion of *relative smoothness* (Van Nguyen, 2017; Bauschke et al., 2016; Lu et al., 2018). Thus, every iteration of the subsolver can be represented as the Gradient Step for the auxiliary problem, with a specific choice of prox-function, formed by the second derivative of the initial objective and augmented by the fourth power of Euclidean norm. The methods of this type were studied in (Grapiglia & Nesterov, 2019b; Bullins, 2018; Nesterov, 2019b).

Up to our knowledge, the efficient implementation of the tensor step of degree $p \geq 4$ remains to be an open question.

3.2. Monotone Inexact Methods

Let us assume that at every step of our method, we minimize the model (4) inexactly by an auxiliary subroutine, up to some given accuracy $\delta \geq 0$. We use the following definition of *inexact δ-step.*

Definition 1 *Denote by $T_{H,\delta}(x)$ a point $T \equiv T_{H,\delta}(x) \in \operatorname{dom}\psi$, satisfying*

$$\Omega_H(x;T) - \min_y \Omega_H(x;y) \quad \leq \quad \delta. \tag{6}$$

The main property of this point is given by the next lemma.

Lemma 1 *Let $H = \alpha L_p$ for some $\alpha \geq p$. Then, for every $y \in \operatorname{dom}\psi$*

$$F(T_{H,\delta}(x)) \quad \leq \quad F(y) + \tfrac{(\alpha+1)L_p\|y-x\|^{p+1}}{(p+1)!} + \delta. \tag{7}$$

Proof:

Indeed, denoting $T \equiv T_{H,\delta}(x)$, we have

$$F(T) \overset{(3)}{\leq} \Omega_H(x;T) \overset{(6)}{\leq} \Omega_H(x;y) + \delta,$$

$$\overset{(3)}{\leq} F(y) + \tfrac{(\alpha+1)L_p\|y-x\|^{p+1}}{(p+1)!} + \delta.$$

□

Now, if we plug $y := x$ (a current iterate) into (7), we obtain $F(T_{H,\delta}(x)) \leq F(x) + \delta$. So in the case $\delta = 0$ (exact tensor step), we would have nonincreasing sequence $\{F(x_k)\}_{k\geq 0}$ of test points of the method. However, this is not the case for $\delta > 0$ (inexact tensor step). Therefore we propose the following minimization scheme *with correction*.

If at some step $k \geq 0$ of this algorithm we get $x_{k+1} = x_k$, then we need to decrease inner accuracy for the next step. From the practical point of view, an efficient implementation of this algorithm should include a possibility of improving accuracy of the previously computed point.

Algorithm 1 Monotone Inexact Tensor Method, I

Initialization: Choose $x_0 \in \operatorname{dom}\psi$. Fix $H := pL_p$.
for $k = 0, 1, 2, \ldots$ **do**
  Pick up $\delta_{k+1} \geq 0$
  Compute inexact tensor step $T_{k+1} := T_{H,\delta_{k+1}}(x_k)$
  if $F(T_{k+1}) < F(x_k)$ **then**
    $x_{k+1} := T_{k+1}$
  else
    $x_{k+1} := x_k$
  end if
end for

Denote by D the radius of the initial level set of the objective:

$$D \quad \overset{\text{def}}{=} \quad \sup_x \Big\{ \|x - x^*\| \ : \ F(x) \leq F(x_0) \Big\}. \tag{8}$$

For Algorithm 1, we can prove the following convergence result, which uses a simple strategy for choosing δ_{k+1}.

Theorem 1 *Let $D < +\infty$. Let the sequence of inner accuracies $\{\delta_k\}_{k\geq 1}$ be chosen according to the rule*

$$\boxed{\delta_k \quad := \quad \tfrac{c}{k^{p+1}}} \tag{9}$$

with some $c \geq 0$. Then for the sequence $\{x_k\}_{k\geq 1}$ produced by Algorithm 1, we have

$$F(x_k) - F^* \quad \leq \quad \tfrac{(p+1)^{p+1}L_pD^{p+1}}{p!\,k^p} + \tfrac{c}{k^p}. \tag{10}$$

We see, that the global convergence rate of the inexact Tensor Method remains on the same level, as of the exact one. Namely, in order to achieve $F(x_K) - F^* \leq \varepsilon$, we need to perform $K = O(1/\varepsilon^{\frac{1}{p}})$ iterations of the algorithm. According to these estimates, at the last iteration K, the rule (9) requires to solve the subproblem up to accuracy

$$\delta_K \quad = \quad O(c\varepsilon^{\frac{p+1}{p}}). \tag{11}$$

This is intriguing, since for bigger p (order of the method) we need less accurate solutions. Note, that the estimate (11) coincides with the constant choice of inner accuracy in (Nesterov, 2019b). However, the dynamic strategy (9) provides a significant decrease of the computational time on the first iterations of the method, which is also confirmed by our numerical results (see Section 5).

Now, looking at Algorithm 1, one may think that we are forgetting the points T_{k+1} such that $F(T_{k+1}) \geq F(x_k)$, and thus we are loosing some computations. However, this is not true: even if point T_{k+1} has not been taken as x_{k+1}, we

shall use it internally as a starting point for computing the next T_{k+2}. To support this concept, we introduce *inexact δ-step* with an additional condition of *monotonicity*.

Definition 2 *Denote by $M_{H,\delta}(x)$ a point $M \equiv M_{H,\delta}(x) \in \operatorname{dom}\psi$, satisfying the following two conditions.*

$$\Omega_H(x;M) - \min_y \Omega_H(x;y) \;\le\; \delta, \tag{12}$$

$$F(M) \;<\; F(x). \tag{13}$$

It is clear, that point M from Definition 2 satisfies Definition 1 as well (while the opposite is not always the case). Therefore, we can also use Lemma 1 for the *monotone* inexact tensor step. Using this definition, we simplify Algorithm 1 and present the following scheme.

Algorithm 2 Monotone Inexact Tensor Method, II

Initialization: Choose $x_0 \in \operatorname{dom}\psi$. Fix $H := pL_p$.
for $k = 0, 1, 2, \dots$ **do**
  Pick up $\delta_{k+1} \ge 0$
  Compute inexact monotone tensor step $x_{k+1} := M_{H,\delta_{k+1}}(x_k)$
end for

When our method is strictly monotone, we guarantee that $F(x_{k+1}) < F(x_k)$ for all $k \ge 0$, and we propose to use the following *adaptive strategy* of defining the inner accuracies.

Theorem 2 *Let $D < +\infty$. Let sequence of inner accuracies $\{\delta_k\}_{k\ge1}$ be chosen in accordance to the rule*

$$\boxed{\delta_k \;:=\; c\cdot\big(F(x_{k-2}) - F(x_{k-1})\big), \quad k \ge 2} \tag{14}$$

for some fixed $0 \le c < \frac{1}{(p+2)3^{p+1}-1}$ and $\delta_1 \ge 0$. Then for the sequence $\{x_k\}_{k\ge1}$ produced by Algorithm 2, we have

$$F(x_k) - F^* \;\le\; \tfrac{\gamma L_p D^{p+1}}{p!\,k^p} + \tfrac{\beta}{k^{p+2}}, \tag{15}$$

where γ and β are the constants:

$$\gamma \;\overset{\text{def}}{=}\; \tfrac{(p+2)^{p+1}}{1-c((p+2)3^{p+1}-1)}, \quad \beta \;\overset{\text{def}}{=}\; \tfrac{\delta_1 + c2^{p+2}(F(x_0)-F^*)}{1-c((p+2)^2/(p+1)-1)}.$$

The rule (14) is surprisingly simple and natural: while the method is approaching the optimum, it becomes more and more difficult to optimize the function. Consequently, the progress in the function value at every step is decreasing. Therefore, we need to solve the auxiliary problem more accurately, and this is exactly what we are doing in accordance to this rule.

It is also notable, that the rule (14) is *universal*, in a sense that it remains the same (up to a constant factor) for the methods of *any order*, starting from $p = 1$.

This strategy also works for the nondegenerate case. Let us assume that our objective is *uniformly convex* of degree $p+1$ with constant σ_{p+1}. Thus, for all $x, y \in \operatorname{dom}\psi$ and $F'(x) \in \partial F(x)$ it holds

$$\begin{aligned} &F(y) - F(x) + \langle F'(x), y - x\rangle \\ &\quad\ge\; \tfrac{\sigma_{p+1}}{p+1}\|y-x\|^{p+1}. \end{aligned} \tag{16}$$

For $p = 1$ this definition corresponds to the standard class of *strongly convex* functions. One of the main sources of uniform convexity is a regularization by power of Euclidean norm (we use this construction in Section 4, where we accelerate our methods):

Example 4 *Let $\psi(x) = \frac{\mu}{p+1}\|x - x_0\|^{p+1}$, $\mu \ge 0$. Then (16) holds with $\sigma_{p+1} = \mu 2^{1-p}$ (see Lemma 5 in (Doikov & Nesterov, 2019c)).*

Example 5 *Let $\psi(x) = \frac{\mu}{2}\|x - x_0\|^2$, $\mu \ge 0$. Consider the ball of radius D around the optimum: $\mathcal{B} = \{x : \|x - x^*\| \le D\}$. Then (16) holds for all $x, y \in \mathcal{B}$ with $\sigma_{p+1} = \frac{(p+1)\mu}{2^p D^{p-1}}$ (see Lemma 5 in the supplementary material).*

Denote by ω_p the *condition number* of degree p:

$$\omega_p \;\overset{\text{def}}{=}\; \max\{\tfrac{(p+1)^2 L_p}{p!\,\sigma_{p+1}}, 1\}. \tag{17}$$

The next theorem shows, that ω_p serves as the main factor in the complexity of solving the uniformly convex problems by inexact Tensor Methods.

Theorem 3 *Let $\sigma_{p+1} > 0$. Let sequence of inner accuracies $\{\delta_k\}_{k\ge1}$ be chosen in accordance to the rule*

$$\boxed{\delta_k \;:=\; c\cdot\big(F(x_{k-2}) - F(x_{k-1})\big), \quad k \ge 2} \tag{18}$$

for some fixed $0 \le c < \frac{p}{p+1}\omega_p^{-1/p}$ and $\delta_1 \ge 0$. Then for the sequence $\{x_k\}_{k\ge1}$ produced by Algorithm 2, we have the following linear rate of convergence:

$$\begin{aligned} &F(x_{k+1}) - F^* \\ &\quad\le\; \Big(1 - \tfrac{p}{p+1}\omega_p^{-1/p} + c\Big)(F(x_{k-1}) - F^*). \end{aligned} \tag{19}$$

Let us pick $c := \frac{p}{2(p+1)}\omega_p^{-1/p}$. Then, according to estimate (19), in order solve the problem up to ε accuracy: $F(x_K) - F^* \le \varepsilon$, we need to perform

$$K \;=\; O\left(\omega_p^{1/p} \log \tfrac{F(x_0)-F^*}{\varepsilon}\right) \tag{20}$$

iterations of the algorithm.

Finally, we study the local behavior of the method for strongly convex objective.

Theorem 4 *Let $\sigma_2 > 0$. Let sequence of inner accuracies $\{\delta_k\}_{k\geq 1}$ be chosen in accordance to the rule*

$$\boxed{\delta_k \;:=\; c\cdot\big(F(x_{k-2})-F(x_{k-1})\big)^{\frac{p+1}{2}},\ k\geq 2} \tag{21}$$

with some fixed $c \geq 0$ and $\delta_1 \geq 0$. Then for $p \geq 2$ the sequence $\{x_k\}_{k\geq 1}$ produced by Algorithm 2 has the local superlinear rate of convergence:

$$\begin{aligned} &F(x_{k+1})-F^* \\ &\leq \Big(\tfrac{L_p}{p!}\big(\tfrac{2}{\sigma_2}\big)^{\frac{p+1}{2}}+c\Big)\big(F(x_{k-1})-F^*\big)^{\frac{p+1}{2}}. \end{aligned} \tag{22}$$

Let us assume for simplicity, that the constant c is chosen to be small enough: $c \leq \frac{L_p}{p!}\big(\frac{2}{\sigma_2}\big)^{(p+1)/2}$. Then, we are able to describe the region of superlinear convergence as

$$\mathcal{Q} = \left\{x \in \operatorname{dom}\psi : F(x)-F^* \leq \left(\frac{\sigma_2^{p+1}}{2^{p+3}}\big(\frac{p!}{L_p}\big)^2\right)^{\frac{1}{(p-1)}}\right\}.$$

After reaching it, the method becomes very fast: we need to perform no more than $O(\log\log\frac{1}{\varepsilon})$ additional iterations to solve the problem.

Note, that estimate (22) of the local convergence is slightly weaker, than the corresponding one for *exact* Tensor Methods (Doikov & Nesterov, 2019b). For example, for $p = 2$ (Cubic regularization of Newton Method) we obtain the convergence of order $\frac{3}{2}$, not the quadratic, which affects only a constant factor in the complexity estimate. The region $\mathcal{Q}$ of the superlinear convergence is remaining the same.

3.3. Inexact Methods with Averaging

Methods from the previous section were developed by forcing the monotonicity of the sequence of function values $\{F(x_k)\}_{k\geq 0}$ into the scheme. As a byproduct, we get the radius of the initial level set D (see definition (8)) in the right-hand side of our complexity estimates (10) and (15). Note, that D may be significantly bigger than the distance $\|x_0 - x^*\|$ from the initial point to the solution.

Example 6 *Consider the following function, for $x \in \mathbb{R}^n$:*

$$f(x) \;=\; |x^{(1)}|^{p+1} + \sum_{i=2}^{n} |x^{(i)} - 2x^{(i-1)}|^{p+1},$$

where $x^{(i)}$ indicates ith coordinate of x. Clearly, the minimum of f is at the origin: $x^* = (0,\dots,0)^T$. Let us take two points: $x_0 = (1,\dots,1)^T$ and x_1, such that $x_1^{(i)} = 2^i - 1$. It holds, $f(x_0) = f(x_1) = n$, so they belong to the same level set. However, we have (for the standard Euclidean norm): $\|x_0 - x^*\| = \sqrt{n}$, while $D \geq \|x_1 - x^*\| \geq 2^{n-1}$.

Here we present an alternative approach, Tensor Methods with *Averaging*. In this scheme, we perform a step not from the previous point x_k, but from a point y_k, which is a convex combination of the previous point and the starting point:

$$y_k \;=\; \lambda_k x_k + (1-\lambda_k)x_0,$$

where $\lambda_k \equiv \big(\frac{k}{k+1}\big)^{p+1}$. The whole optimization scheme remains very simple.

Algorithm 3 Inexact Tensor Method with Averaging

Initialization: Choose $x_0 \in \operatorname{dom}\psi$. Fix $H := pL_p$.
for $k = 0, 1, 2, \dots$ **do**
  Set $\lambda_k := \big(\frac{k}{k+1}\big)^{p+1}$, $y_k := \lambda_k x_k + (1-\lambda_k)x_0$
  Pick up $\delta_{k+1} \geq 0$
  Compute inexact tensor step $x_{k+1} := T_{H,\delta_{k+1}}(y_k)$
end for

For this method, we are able to prove a similar convergence result as that of Algorithm 1. However, now we have the explicit distance $\|x_0 - x^*\|$ in the right hand side of our bound for the convergence rate (compare with Theorem 1).

Theorem 5 *Let sequence of inner accuracies $\{\delta_k\}_{k\geq 1}$ be chosen according the rule*

$$\boxed{\delta_k \;:=\; \tfrac{c}{k^{p+1}}} \tag{23}$$

for some $c \geq 0$. Then for the sequence $\{x_k\}_{k\geq 1}$ produced by Algorithm 3, we have

$$F(x_k) - F^* \;\leq\; \tfrac{(p+1)^{p+1}L_p\|x_0-x^*\|^{p+1}}{p!\,k^p} + \tfrac{c}{k^p}. \tag{24}$$

Thus, Algorithm 3 seems to be the first *Primal* Tensor method (aggregating only the points from the primal space $\mathbb{E}$), which admits the explicit initial distance in the global convergence estimate (24). Table 1 contains a short overview of the inexact Tensor methods from this section.

Table 1. Summary on the methods.

Algorithm	The rule for δ_k	Global rate	Local superl.
Tensor Method (Nesterov, 2019a)	0	$O\big(\frac{L_p D^{p+1}}{k^p}\big)$	Yes
Monotone Inexact Tensor Method, I (Algorithm 1)	$1/k^{p+1}$	$O\big(\frac{L_p D^{p+1}}{k^p}\big)$	No
Monotone Inexact Tensor Method, II (Algorithm 2)	$(F(x_{k-1})-F(x_k))^\alpha$	$O\big(\frac{L_p D^{p+1}}{k^p}\big)$, $\alpha = 1$	Yes, $\alpha = \frac{p+1}{2}$
Inexact Tensor Method with Averaging (Algorithm 3)	$1/k^{p+1}$	$O\big(\frac{L_p\|x_0-x^*\|^{p+1}}{k^p}\big)$	No

4. Acceleration

After the Fast Gradient Method had been discovered in (Nesterov, 1983), there were made huge efforts to develop accelerated second-order (Nesterov, 2008; Monteiro & Svaiter, 2013; Grapiglia & Nesterov, 2019a) and high-order (Baes, 2009; Nesterov, 2019a; Gasnikov et al., 2019; Grapiglia & Nesterov, 2019c; Song & Ma, 2019) optimization algorithms. Most of these schemes are based on the notion of Estimating sequences (see (Nesterov, 2018)).

An alternative approach of using the Proximal Point iterations with the acceleration was studied first in (Güler, 1992). It became very popular recently, in the context of machine learning applications (Lin et al., 2015; 2018; Kulunchakov & Mairal, 2019; Ivanova et al., 2019). In this section, we use the technique of Contracting Proximal iterations (Doikov & Nesterov, 2019a), to accelerate our inexact tensor methods.

In the accelerated scheme, two sequences of points are used: the main sequence $\{x_k\}_{k\geq 0}$, for which we are able to guarantee the convergence in function residuals, and auxiliary sequence $\{v_k\}_{k\geq 0}$ of prox-centers, starting from the same initial point: $v_0 = x_0$. Also, we use the sequence $\{A_k\}_{k\geq 0}$ of scaling coefficients. Denote $a_k \overset{\text{def}}{=} A_k - A_{k-1}, k \geq 1$.

Then, at every iteration, we apply Monotone Inexact Tensor Method, II (Algorithm 2) to minimize the following *contracted* objective with regularization:

$$h_{k+1}(x) \overset{\text{def}}{=} A_{k+1} f\left(\tfrac{a_{k+1}x + A_k x_k}{A_{k+1}}\right) + a_{k+1}\psi(x) + \beta_d(v_k; x).$$

Here $\beta_d(v_k; x) \overset{\text{def}}{=} d(x) - d(v_k) - \langle \nabla d(v_k), x - v_k \rangle$ is *Bregman divergence* centered at v_k, for the following choice of prox-function:

$$d(x) := \tfrac{1}{p+1}\|x - x_0\|^{p+1},$$

which is uniformly convex of degree $p+1$ (Example 4). Therefore, Tensor Method achieves fast linear rate of convergence (Theorem 3). By an appropriate choice of scaling coefficients $\{A_k\}_{k\geq 1}$, we are able to make the condition number of the subproblem being an absolute constant. This means that only $\tilde{O}(1)$ steps of Algorithm 2 are needed to find an approximate minimizer of $h_{k+1}(\cdot)$:

$$h_{k+1}(v_{k+1}) - h^*_{k+1} \leq \zeta_{k+1}. \tag{25}$$

Note that inexact condition (25) was considered first in (Lin et al., 2015), in a general algorithmic framework for accelerating first-order methods. It differs from the corresponding one from (Doikov & Nesterov, 2019a), where a bound for the (sub)gradient norm was used.

Therefore, for accelerating inexact Tensor Methods, we propose a multi-level approach. On the upper level, we run Algorithm 4. At each iteration of this method, we call Algorithm 2 (with dynamic rule (18) for inner accuracies) to find v_{k+1}. For this optimization scheme, we obtain the following global convergence guarantee.

Algorithm 4 Accelerated Scheme

Initialization: Choose $x_0 \in \operatorname{dom}\psi$. Set $v_0 := x_0$, $A_0 := 0$.
for $k = 0, 1, 2, \ldots$ **do**
  Set $A_{k+1} := \frac{(k+1)^{p+1}}{L_p}$
  Pick up $\zeta_{k+1} \geq 0$
  Find v_{k+1} such that (25) holds
  Set $x_{k+1} := \frac{a_{k+1}v_{k+1} + A_k x_k}{A_{k+1}}$
end for

Theorem 6 *Let sequence $\{\zeta_k\}_{k\geq 1}$ be chosen according to the rule*

$$\boxed{\zeta_k := \tfrac{c}{k^{p+2}}} \tag{26}$$

with some $c \geq 0$. Then for the iterations $\{x_k\}_{k\geq 1}$ produced by Algorithm 4, it holds:

$$F(x_k) - F^* \leq O\left(\tfrac{L_p(\|x_0 - x^*\|^{p+1} + c)}{k^{p+1}}\right). \tag{27}$$

For every $k \geq 0$, in order to find v_{k+1} by Algorithm 2 (for minimizing $h_{k+1}(\cdot)$, starting from v_k), it is enough to perform no more than

$$O\left(\log \tfrac{(k+1)(\|x_0 - x^*\|^{p+1} + c)}{c}\right). \tag{28}$$

inexact monotone tensor steps.

Therefore, the total number of the inexact tensor steps for finding ε-solution of (1) is bounded by $\widetilde{O}\left(1/\varepsilon^{\frac{1}{p+1}}\right)$. One theoretical question remains open: is it possible to construct in the framework of inexact tensor steps, the *optimal methods* with the complexity estimate $O\left(1/\varepsilon^{\frac{2}{3p+1}}\right)$ having no hidden logarithmic factors. This would match the existing lower bound (Arjevani et al., 2019; Nesterov, 2019a).

5. Experiments

Let us demonstrate computational results with empirical study of different accuracy policies. We consider inexact methods of order $p = 2$ (Cubic regularization of Newton method), and to solve the corresponding subproblem we call the Fast Gradient Method with restarts from (Nesterov, 2019b). To estimate the residual in function value of the subproblem, we use a simple stopping criterion, given by uniform convexity of the model $g(y) = \Omega_H(x; y)$:

$$g(y) - \min_y g(y) \leq \tfrac{4}{3}\left(\tfrac{1}{H}\right)^{1/2} \|\nabla g(y)\|_*^{3/2}. \tag{29}$$

An alternative approach would be to bound the functional residual by the duality gap[1].

We compare the *adaptive* rule for inner accuracies (14) with dynamic strategies in the form $\delta_k = 1/k^\alpha$, for different α (left graphs), and with the constant choices (right).

5.1. Logistic Regression

First, let us consider the problem of training ℓ_2-regularized logistic regression model for classification task with two classes, on several real datasets[2]: *mashrooms* ($m = 8124, n = 112$), *w8a* ($m = 49749, n = 300$), and *a8a* ($m = 22696, n = 123$)[3].

We use the standard Euclidean norm for this problem, and simple line search at every iteration, to fit the regularization parameter H. The results are shown on Figure 1.

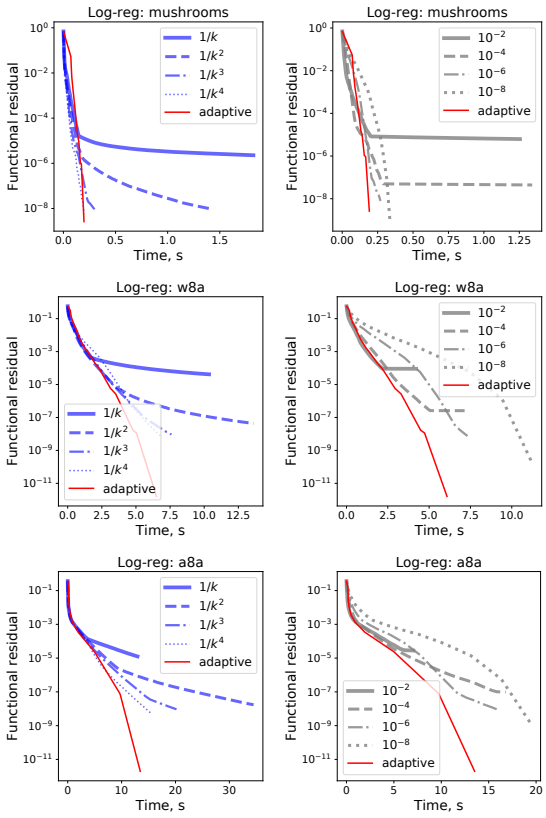

Figure 1. Comparison of different accuracy policies for inexact Cubic Newton, training logistic regression.

[1]Note that the left hand side in (29) can be bounded from below by the distance from y to the optimum of the model, using uniform convexity. Therefore, we have a computable bound for the distance to the solution of the subproblem.

[2]https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

[3]m is the number of training examples and n is the dimension of the problem (the number of features).

5.2. Log-Sum-Exp

In the next set of experiments, we consider unconstrained minimization of the following objective:

$$f_\mu(x) \quad = \quad \mu \ln \left(\sum_{i=1}^m \exp\left(\frac{\langle a_i, x \rangle - b_i}{\mu} \right) \right), \quad x \in \mathbb{R}^n,$$

where $\mu > 0$ is a *smoothing* parameter. To generate the data, we sample coefficients $\{\tilde{a}_i\}_{i=1}^m$ and b randomly from the uniform distribution on $[-1, 1]$. Then, we shift the parameters in a way to have the solution x^* in the origin. Namely, using $\{\tilde{a}_i\}_{i=1}^m$ we form a preliminary function $\tilde{f}_\mu(x)$, and set $a_i := \tilde{a}_i - \nabla \tilde{f}_\mu(0)$. Thus we essentially obtain $\nabla f_\mu(0) = 0$.

We set $m = 6n$, and $n = 100$. In the method, we use the following Euclidean norm for the primal space: $\|x\| = \langle Bx, x \rangle^{1/2}$, with the matrix $B = \sum_{i=1}^m a_i a_i^T$, and fix regularization parameter H being equal 1. The results are shown on Figure 2.

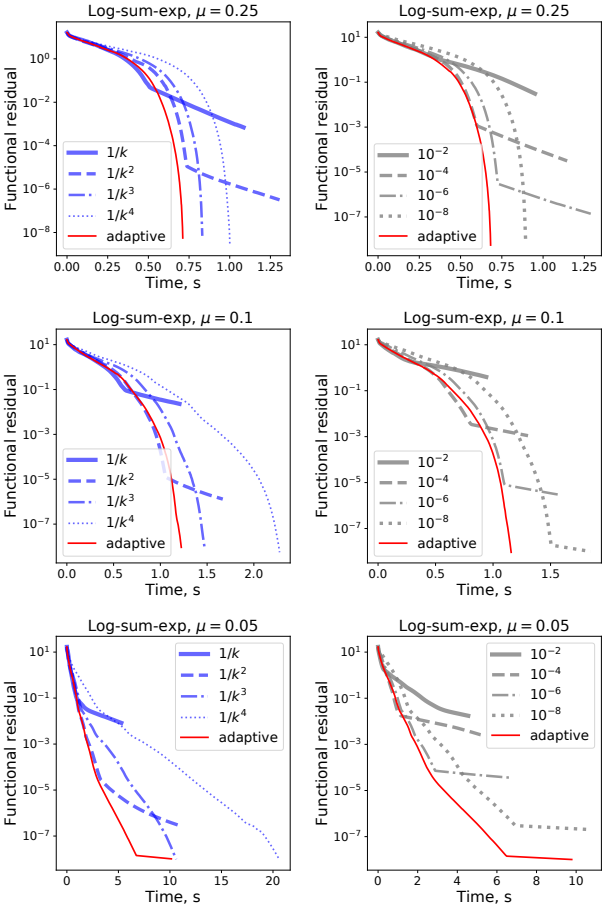

Figure 2. Comparison of different accuracy policies for inexact Cubic Newton, minimizing Log-Sum-Exp function.

We see, that the adaptive rule demonstrates reasonably good

performance (in terms of the total computational time[4]) in all the scenarios.

Acknowledgements

The research results of this paper were obtained in the framework of ERC Advanced Grant 788368.

References

Agarwal, N., Allen-Zhu, Z., Bullins, B., Hazan, E., and Ma, T. Finding approximate local minima faster than gradient descent. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 1195–1199. ACM, 2017.

Arjevani, Y., Shamir, O., and Shiff, R. Oracle complexity of second-order methods for smooth convex optimization. *Mathematical Programming*, 178(1-2):327–360, 2019.

Baes, M. Estimate sequence methods: extensions and approximations. *Institute for Operations Research, ETH, Zürich, Switzerland*, 2009.

Bauschke, H. H., Bolte, J., and Teboulle, M. A descent lemma beyond lipschitz gradient continuity: first-order methods revisited and applications. *Mathematics of Operations Research*, 42(2):330–348, 2016.

Beck, A. *First-order methods in optimization*, volume 25. SIAM, 2017.

Birgin, E. G., Gardenghi, J., Martínez, J. M., Santos, S. A., and Toint, P. L. Worst-case evaluation complexity for unconstrained nonlinear optimization using high-order regularized models. *Mathematical Programming*, 163 (1-2):359–368, 2017.

Bullins, B. Fast minimization of structured convex quartics. *arXiv preprint arXiv:1812.10349*, 2018.

Carmon, Y. and Duchi, J. Gradient descent finds the cubic-regularized nonconvex Newton step. *SIAM Journal on Optimization*, 29(3):2146–2178, 2019.

Cartis, C. and Scheinberg, K. Global convergence rate analysis of unconstrained optimization methods based on probabilistic models. *Mathematical Programming*, 169 (2):337–375, 2018.

Cartis, C., Gould, N. I., and Toint, P. L. Adaptive cubic regularisation methods for unconstrained optimization. Part I: motivation, convergence and numerical results. *Mathematical Programming*, 127(2):245–295, 2011a.

Cartis, C., Gould, N. I., and Toint, P. L. Adaptive cubic regularisation methods for unconstrained optimization. Part II: worst-case function-and derivative-evaluation complexity. *Mathematical programming*, 130(2):295–319, 2011b.

Cartis, C., Gould, N. I., and Toint, P. L. Universal regularization methods: varying the power, the smoothness and the accuracy. *SIAM Journal on Optimization*, 29(1): 595–615, 2019.

Doikov, N. and Nesterov, Y. Contracting proximal methods for smooth convex optimization. *CORE Discussion Papers 2019/27*, 2019a.

Doikov, N. and Nesterov, Y. Local convergence of tensor methods. *CORE Discussion Papers 2019/21*, 2019b.

Doikov, N. and Nesterov, Y. Minimizing uniformly convex functions by cubic regularization of Newton method. *arXiv preprint arXiv:1905.02671*, 2019c.

Doikov, N. and Richtárik, P. Randomized block cubic Newton method. In *International Conference on Machine Learning*, pp. 1289–1297, 2018.

Gasnikov, A., Dvurechensky, P., Gorbunov, E., Vorontsova, E., Selikhanovych, D., Uribe, C. A., Jiang, B., Wang, H., Zhang, S., Bubeck, S., et al. Near optimal methods for minimizing convex functions with lipschitz p-th derivatives. In *Conference on Learning Theory*, pp. 1392–1393, 2019.

Gould, N. I., Robinson, D. P., and Thorne, H. S. On solving trust-region and other regularised subproblems in optimization. *Mathematical Programming Computation*, 2 (1):21–57, 2010.

Grapiglia, G. N. and Nesterov, Y. Accelerated regularized Newton methods for minimizing composite convex functions. *SIAM Journal on Optimization*, 29(1):77–99, 2019a.

Grapiglia, G. N. and Nesterov, Y. On inexact solution of auxiliary problems in tensor methods for convex optimization. *arXiv preprint arXiv:1907.13023*, 2019b.

Grapiglia, G. N. and Nesterov, Y. Tensor methods for minimizing functions with Hölder continuous higher-order derivatives. *arXiv preprint arXiv:1904.12559*, 2019c.

Grapiglia, G. N. and Nesterov, Y. Tensor methods for finding approximate stationary points of convex functions. *arXiv preprint arXiv:1907.07053*, 2019d.

Güler, O. New proximal point algorithms for convex minimization. *SIAM Journal on Optimization*, 2(4):649–664, 1992.

[4]Clock time was evaluated using the machine with Intel Core i5 CPU, 1.6GHz; 8 GB RAM. All methods were implemented in Python. The source code can be found at https://github.com/doikov/dynamic-accuracies/

Ivanova, A., Grishchenko, D., Gasnikov, A., and Shulgin, E. Adaptive catalyst for smooth convex optimization. *arXiv preprint arXiv:1911.11271*, 2019.

Jiang, B., Lin, T., and Zhang, S. A unified adaptive tensor approximation scheme to accelerate composite convex optimization. *arXiv preprint arXiv:1811.02427*, 2018.

Kohler, J. M. and Lucchi, A. Sub-sampled cubic regularization for non-convex optimization. In *International Conference on Machine Learning*, pp. 1895–1904, 2017.

Kulunchakov, A. and Mairal, J. A generic acceleration framework for stochastic composite optimization. In *Advances in Neural Information Processing Systems*, pp. 12556–12567, 2019.

Lin, H., Mairal, J., and Harchaoui, Z. A universal catalyst for first-order optimization. In *Advances in Neural Information Processing Systems*, pp. 3384–3392, 2015.

Lin, H., Mairal, J., and Harchaoui, Z. Catalyst acceleration for first-order convex optimization: from theory to practice. *Journal of Machine Learning Research*, 18(212): 1–54, 2018.

Lu, H., Freund, R. M., and Nesterov, Y. Relatively smooth convex optimization by first-order methods, and applications. *SIAM Journal on Optimization*, 28(1):333–354, 2018.

Lucchi, A. and Kohler, J. A stochastic tensor method for non-convex optimization. *arXiv preprint arXiv:1911.10367*, 2019.

Monteiro, R. D. and Svaiter, B. F. An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods. *SIAM Journal on Optimization*, 23(2):1092–1125, 2013.

Nesterov, Y. A method for solving the convex programming problem with convergence rate O(1/kˆ2). In *Dokl. akad. nauk Sssr*, volume 269, pp. 543–547, 1983.

Nesterov, Y. Accelerating the cubic regularization of Newton's method on convex problems. *Mathematical Programming*, 112(1):159–181, 2008.

Nesterov, Y. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140(1):125–161, 2013.

Nesterov, Y. *Lectures on convex optimization*, volume 137. Springer, 2018.

Nesterov, Y. Implementable tensor methods in unconstrained convex optimization. *Mathematical Programming*, pp. 1–27, 2019a.

Nesterov, Y. Inexact basic tensor methods. *CORE Discussion Papers 2019/23*, 2019b.

Nesterov, Y. and Nemirovskii, A. *Interior-point polynomial algorithms in convex programming*. SIAM, 1994.

Nesterov, Y. and Polyak, B. T. Cubic regularization of Newton's method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

Nocedal, J. and Wright, S. J. Numerical optimization 2nd, 2006.

Rodomanov, A. and Nesterov, Y. Smoothness parameter of power of euclidean norm. *arXiv preprint arXiv:1907.12346*, 2019.

Schmidt, M., Roux, N. L., and Bach, F. R. Convergence rates of inexact proximal-gradient methods for convex optimization. In *Advances in neural information processing systems*, pp. 1458–1466, 2011.

Song, C. and Ma, Y. Towards unified acceleration of high-order algorithms under Hölder continuity and uniform convexity. *arXiv preprint arXiv:1906.00582*, 2019.

Tripuraneni, N., Stern, M., Jin, C., Regier, J., and Jordan, M. I. Stochastic cubic regularization for fast nonconvex optimization. In *Advances in Neural Information Processing Systems*, pp. 2899–2908, 2018.

Van Nguyen, Q. Forward-backward splitting with bregman distances. *Vietnam Journal of Mathematics*, 45(3):519–539, 2017.

Wang, Z., Zhou, Y., Liang, Y., and Lan, G. Stochastic variance-reduced cubic regularization for nonconvex optimization. *arXiv preprint arXiv:1802.07372*, 2018.

Zhou, D., Xu, P., and Gu, Q. Stochastic variance-reduced cubic regularization methods. *Journal of Machine Learning Research*, 20(134):1–47, 2019.